APPLICATION

FOR

UNITED STATES LETTERS PATENT

TITLE: BIOS UPDATE FILE

APPLICANT: MATTHEW D. SINGER, ROBERT J. JOHNSON AND NICHOLAS J. ADAMS

CERTIFICATE OF MAILING BY EXPRESS MAIL

Express Mail Label No. EV 331002132 US

December 4, 2003

Date of Deposit

BIOS UPDATE FILE

TECHNICAL FIELD

The present invention relates to a basic input/output system (BIOS) update file.

BACKGROUND

A basic input/output system (BIOS) is built-in software that determines what a computer can do without accessing programs from a disk. The BIOS contains all the code required to control, for example, a keyboard, a display screen, disk drives, serial communications, and to perform miscellaneous functions.

The BIOS is typically placed in a read only memory (ROM) chip that comes with the computer. Modern computers have a so-called flash BIOS, which means that the BIOS is recorded on a flash memory chip and can be modified using an update provided by, for example, an original equipment manufacturer (OEM), in a BIOS update file on a diskette.

Multiple BIOS update files are used to modify multiple flash memory modules in a system's flash BIOS. For example, a first BIOS update file can modify a core of a system's BIOS. A second BIOS update file can modify a company logo flash BIOS module. A third BIOS update file can modify a language flash BIOS module, e.g., English, with another language, e.g., French.

DESCRIPTION OF DRAWINGS

FIG. 1 is a block diagram.

FIG. 2 is a block diagram.

FIG. 3 is a flow diagram.

DETAILED DESCRIPTION

As shown in FIG. 1, an exemplary third party computing system 10 includes a processor 12 and memory 14, manufactured by an original equipment manufacture (OEM), such as Intel Corporation. The system 10 also includes storage devices 16 and an input/output (I/O) device 18. Example storage devices 16 are disk drives and floppy drives. The I/O device 18 can include a display screen 20 and keyboard 22.

Memory 14 includes an operating system 24 such as Windows XP or Linux, a basic input/output operating system (BIOS) installation process 25 and a flash memory 26 containing the BIOS of system 10.

Flash memory (sometimes called "flash RAM") is a type of constantly-powered nonvolatile memory that can be erased and reprogrammed in units of memory called blocks. It is a variation of electrically erasable programmable read-only memory (EEPROM) which, unlike flash memory, is erased and rewritten at the byte level, which is slower than flash memory updating. Flash memory is often used to hold control code such as the basic input/output system (BIOS) in a personal computer. When BIOS needs to be changed (rewritten), the flash memory can be written to in block (rather than byte) sizes, making it easy to update.

Flash memory gets its name because a section of memory cells are erased in a single action or "flash." The erasure is caused by Fowler-Nordheim tunneling in which electrons pierce through a thin dielectric material to remove an electronic charge from a floating gate associated with each memory cell. Intel Corporation offers a form of flash memory that holds two bits (rather than one) in each memory cell, thus doubling the capacity of memory without a corresponding increase in price.

Flash memory 26 is organized into flash memory modules 28. Flash memory modules 28 contain the BIOS code required to

control, for example, the storage devices 16, the display screen 20, the keyboard 22, serial communications (not shown), and to perform functions, respectively. The system 10 is assembled by a third party, such as Dell Computer Company, with hardware (e.g., storage devices 16 and serial communications) manufactured by other vendors. The third party relies on the OEM for keeping the BIOS current.

The BIOS installation process 25 is provided by the OEM and is used by the third party to install BIOS updates to the BIOS code in the flash memory modules 28 of the flash memory 26. The BIOS updates are secure data residing in a signed BIOS update file, described below, to insure data integrity and prevent misuse.

As shown in FIG. 2, a BIOS update file 50 is generated in a secure fashion by an original equipment manufacturer (OEM), such as Intel Corporation, for installation in a flash memory of targeted hardware included in a third party's system, such as a system assembled by Dell Computer Company. The BIOS update file 50 includes a BIOS file header 52, a signed data portion 54, an unsigned data portion 56 and a signature 59.

The BIOS file header 52 includes interface data in conformance with an extensible firmware interface (EFI) specification. The EFI specification defines a model for an interface between operating systems and platform hardware. The interface includes data tables that contain platform-related information, plus boot and runtime service calls that are available to the operating system and its loader. Together, these provide a standard environment for booting an operating system and running pre-boot applications. The BIOS file header 52 also provides backward and forward compatibility to the BIOS update file 50.

The signed data portion 54 includes a volume header 58, signed data 60 and update code 62 (also referred to as a configuration utility). The volume header 58 contains a list of the locations of everything contained within the BIOS update file 50.

The signed data 60 includes a secure copy of the entire trusted BIOS update data 61 generated by the OEM along with an access control list 63. Portions of the data 61 can be included in or removed from a BIOS image to be inserted into the third party system, such as system 10, by an unauthenticated third party. This same access control list 63 is also embedded in the flash memory 26 of the targeted hardware of the third party's system 10 by the OEM and can be used by the OEM to control installation of different portions of a BIOS during BIOS updates.

The unauthenticated third party uses an OEM developed installation process to configure the data 61 using unsigned data contained in the unsigned data portion 56. The unauthenticated third party communicates with the update code 62 through a graphical user interface (GUI) of the installation process, which provides command and data structures in the unsigned data portion 56. The update code 62 uses the access control list 63 to enforce security rules regarding the types of configuration modifications permitted to the data 61 by the unauthenticated third party. The access control list 63 allows the unauthenticated third party an ability to add, modify and/or delete certain data 61 of the signed data 60 and insert their own data from the unsigned data portion 56 without the OEM losing confidence in the integrity of the signed data 60 that is eventually loaded into the flash memory modules 28 of system 10. For example, the unauthenticated third party can add their company logo or specify a specific language, such as French.

The update code 62 is executable code. The update code 62 processes commands and corresponding data to perform actual configuration modifications to the data 61 that becomes a final image that is placed into the flash memory modules 28 of the unauthenticated third party's system 10. Because the update code 62 is executable, new algorithms can be implemented by the OEM after initial loading of BIOS code in the flash memory modules 28.

The signature 59 is used to authenticate the BIOS update file 50. The signature 59 is verified against a verification algorithm 21 that is embedded in flash memory 26 prior to the flashing of the BIOS during the BIOS update installation process 25. In some implementations, the signature 59 uses public key/private key encryption and RSA algorithms.

The unsigned data portion 56 includes an update command list 66 and unsigned data 68. The list 66 includes instructions provided by the unauthenticated third party during execution of the update code 62 to modify the data 61, i.e., the unsigned data 68 is used in conjunction with the list 66 and replaces some of the data 61. The resulting BIOS update file 50 then includes all trusted BIOS data in a single, digitally signed BIOS update file 50. No other data files contained in other BIOS update files are necessary to change a configuration of BIOS code in the flash memory modules 28 of the unauthenticated third party's system 10. The update code 62 uses the BIOS update file 50 to generate an image that is written (flashed) into the flash memory modules 28. Multiple flash memory modules 28 can be updated by a single BIOS update file 50.

When the OEM initially generates the BIOS update file 50 it contains only the BIOS file header 52, the signed data portion 54, and an empty unsigned data portion 56. The unauthenticated third party, using the update code 62, adds unsigned data 68 to

the unsigned data portion 56 and instructions/commands 66 on how data 68 in the unsigned data portion 56 should be used to modify some of the data 61. During the BIOS update installation process 25, the update code 62 replaces some of the data 61 with data 68 in the unsigned data portion 56 according to instructions 66 contained within the unsigned data portion 56. During the BIOS update installation process 25 and prior to any replacement of the data 61, instructions 66 contained in the unsigned data portion 56 are checked against a list of permitted instructions contained in the access control list 63.

As shown in FIG. 3, a BIOS update process 100 includes an OEM generating (102) a secure BIOS update file. The secure BIOS update file includes a BIOS file header, a signed data portion including executable update code, a signature and an unsigned data portion. The unsigned data portion is empty. An unauthenticated third party configures (104) the unsigned portion of the BIOS update file using executable update code. Modifications include unauthenticated third party data and a list of instructions to be used in conjunction with the unauthenticated third party data.

BIOS update installation is invoked (106) by a BIOS update installation process residing in the memory of the third party system. The process 100 verifies (108) the digital signature in the BIOS update file against a signature embedded in the third party system's flash memory 26. If the digital signature verification fails, the process 100 aborts (110).

If the digital signature is verified, the update code residing in the signed data portion is executed (112). The update code processes any commands in the unsigned data area after verification of the commands against an access control list residing in the signed data portion. Commands that are not permitted are ignored.

When the update code terminates execution the resultant modified signed data portion is committed (114) as a trusted image to a flash memory of the third party's system.

Invoking (106) the BIOS installation process verifies the size of the incoming image relative to the amount of space available in the BIOS. A buffer is applied so that if the modules 28 with the current BIOS grow, future BIOS flashes will still have enough space to be able to work.

The BIOS update file 10 allows an unauthenticated third party that is not the OEM to selectively add, modify and/or remove BIOS components from the secure signed data portion 14 without OEM involvement and without the OEM losing trust in the resulting modified components. All trusted components are included in a single digitally signed BIOS update file 10. No other data files are necessary to change a configuration of a BIOS.

Other embodiments are within the scope of the following claims.